



Thermodynamic model creation

*Prospects and challenges for performing MCMC inversions
on multicomponent data with ENKI*

ENKI Workshop, August 7th 2019

Bob Myhill

Motivation

- ▶ Many of us work in experimental petrology, collecting data to better understand phase relations¹.
- ▶ Existing endmember/solution models are built with closed source tools, and (of course) didn't use our cool new data!
- ▶ We seek to build a robust, flexible, open-source model creator that incorporates data from many different sources, and deals consistently with:
 1. multi-solution, multi-reaction assemblages
 2. (correlated) state and compositional uncertainties
 3. incomplete knowledge of site distributions in solutions

My motivation for this project is to model iron autoredox ($\text{Fe}^{2+} \longleftrightarrow \text{Fe}^{3+} + \text{Fe}^0$) reactions in the mantle transition zone.

Recent datasets: Holland et al., 2018 (tc-ds633)

- ▶ Huge dataset of minerals (289 endmembers) from the crust to the lower mantle
- ▶ Large number of reactions used to invert for endmember properties
- ▶ Large **but incomplete** number of chemical components considered (NCKFMASHTO+Mn-Cr-Ti...)
- ▶ Flexible heat capacity terms in their equation of state *although a tweak is needed to modify the EoS for t.z/lower mantle use*
- ▶ Mostly restricted to fitting standard state enthalpy
- ▶ The endmember inversion is (mostly) separate from the solid solution inversion
- ▶ Covariance matrix included with the dataset ☺

Recent datasets: Stixrude and Lithgow-Bertelloni, 2011

- ▶ Smallish dataset (47 endmembers), mostly mantle minerals
- ▶ Limited but useful chemical system (NCFMAS)
- ▶ Simultaneous (although iterative) inversion of endmember and solution model data ☺
- ▶ No order-disorder in any solid solutions
- ▶ No covariance matrix

We determine the values of the parameters by minimizing the misfit

$$\chi^2 = \sum_i^{\text{stability}} \frac{[\mathcal{G}(P_i, T_i, \bar{n}_i) - \mathcal{G}(P_i, T_i, \bar{n}_i^{\text{min}})]^2}{\sigma_i^2} + \sum_i^{\text{reactions}} \frac{[\Delta\mu_i(P_i, T_i, \bar{n}_i)]^2}{\sigma_i^2}, \quad (\text{A1})$$

We treat the uncertainties in a very simple way. We assign a universal and nominal value of $\sigma = 1 \text{ kJ mol}^{-1} \text{ atom}^{-1}$ to all stability observations.

We assign $\sigma = 3 \text{ kJ mol}^{-1} \text{ atom}^{-1}$ to all reactions, where the greater uncertainty as compared with that assumed for stability observations accounts for uncertainties in phase compositions in addition to those in P .

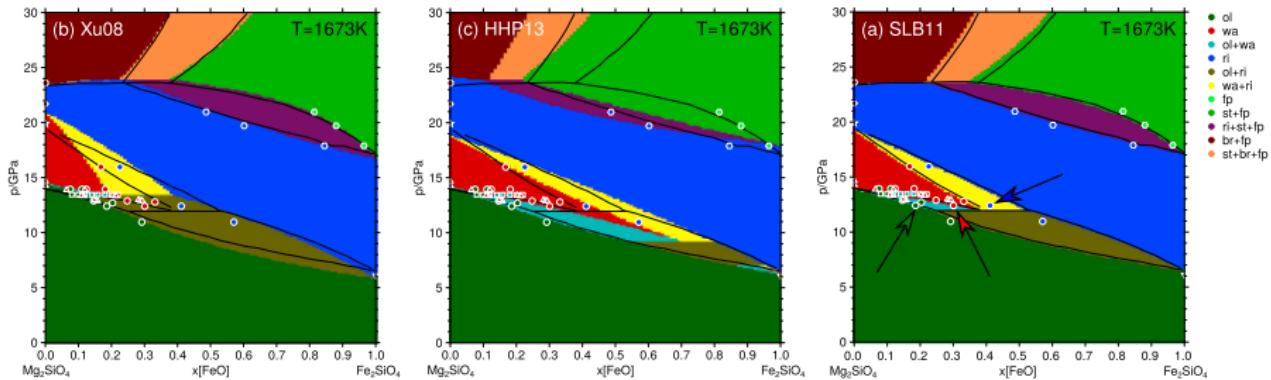


An example of recent discrepancies: Solid solutions of the olivine polymorphs

	fper	ol	wad	ring
Frost (2003)	11.1	2	7.5	4.16
Fabrichnaya et al. (2004)	-	4.5	5.2 -7.2T	16.9 - 16.2T
Stixrude + L-B (2011)	13	3.9	8.37	4.67
Holland et al. (2013)	18	4.5	6.5	2
Jacobs et al. (2019)	13.4 ± 1.0	4.8 ± 0.5	4.43 ± 0.7	3.9 ± 0.6

Fe-Mg interaction energies (one site per formula unit)

An example of recent discrepancies: Solid solutions of the olivine polymorphs



Differences between the Xu et al. (2008), Holland et al. (2013) and Stixrude and Lithgow-Bertelloni (2011) model predictions for the Mg₂SiO₄–Fe₂SiO₄ phase diagram. Black lines are assemblage boundaries from Stixrude and Lithgow-Bertelloni. From Chust et al. (2017).

Part I: Setting up the problem

We want to set up a Monte Carlo simulation to generate a probability distribution of parameter values Θ from an experimental (and *ab-initio*) dataset D .

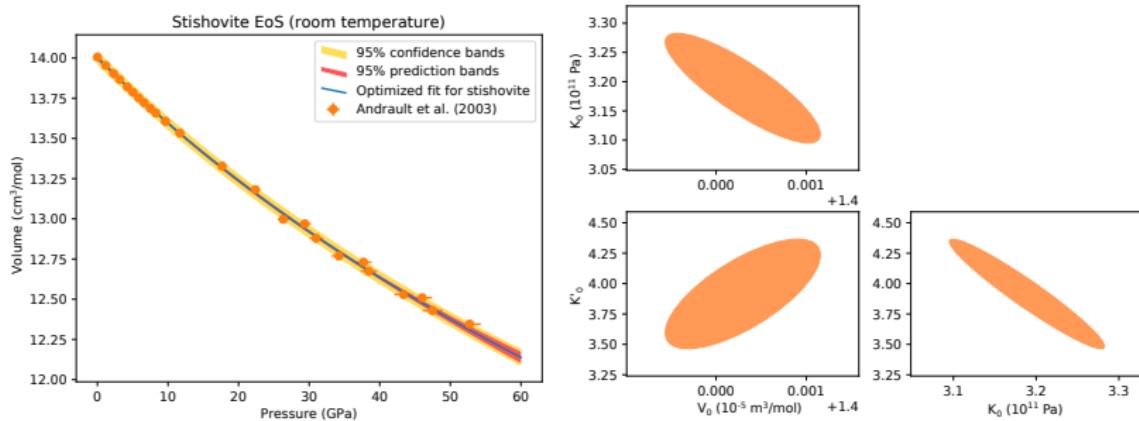
The basis of Bayesian probability theory is to use the following relationship between probabilities:

$$p(\Theta|D) = \frac{p(\Theta)p(D|\Theta)}{p(D)}$$

$p(D)$ is just a normalisation constant, so let's look at how we can construct our prior probability distribution function $p(\Theta)$ and our likelihood function $p(D|\Theta)$.

The prior probability distribution function $p(\Theta)$

For this test problem, we use endmember PTV data and endmember and solution calorimetry data as priors...



Our parameters include standard state H_0 , S_0 , V_0 , a_0 and K_0 for endmembers and E_{ij} , S_{ij} , V_{ij} interactions for solid solutions.

The likelihood function $p(D|\Theta)$

Definition

Given a certain parameter sample Θ_i , how likely is it that we obtained the observed dataset?

This is a non-trivial question - we need to define some sort of misfit that captures how far away we are from equilibrium for each assemblage in the dataset.

- ▶ We **don't** want to calculate an equilibrium state for every experimental run and every parameter sample (this is very expensive, even with ENKI!).
- ▶ We **must** propagate experimental uncertainties through our calculations (the parameter uncertainties may become highly correlated).

The likelihood function $p(D|\Theta)$

We define our likelihood function using the vector of reaction affinities a :

$$a = R \cdot \mu \quad (1)$$

$$\text{Cov}(a) = R \cdot \text{Cov}(\mu) \cdot R^T \quad (2)$$

$$\chi^2 = a \cdot \text{Cov}(a)^{-1} \cdot a^T \quad (3)$$

where R is the reaction matrix, $\mu(X, P, T)$ is the vector of endmember partial gibbs free energies and $\text{Cov}(\mu)$ is the covariance matrix of those free energies. The total probability of the entire dataset $D = \{A\}$ given parameters Θ is then given by the following sum:

$$\ln(p(D|\Theta)) = \sum_i \left(\frac{\chi^2}{2} \right) \quad (4)$$

To use the above, R , μ and $\text{Cov}(\mu)$ must be defined for each assemblage.

Finding R , μ and $\text{Cov}(\mu)$: R

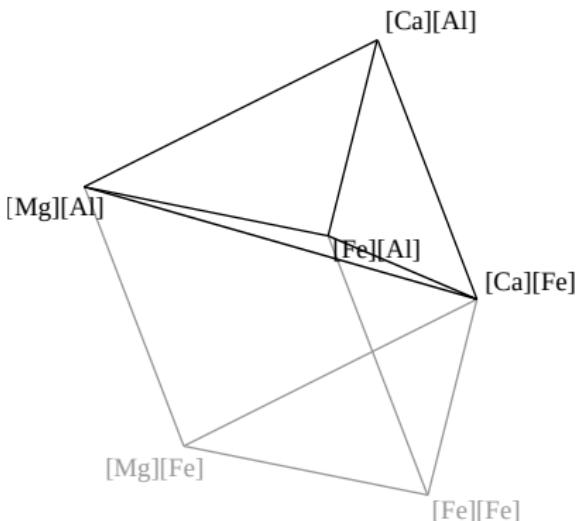
- ▶ The reaction matrix should contain only the endmember phases which play an important rôle in reactions.
- ▶ For example, garnets in Cr-poor rocks will typically have negligible uvarovite contents.
- ▶ Therefore, uvarovite should not be included as a potential endmember when calculating R .



©Lina Jakaitė

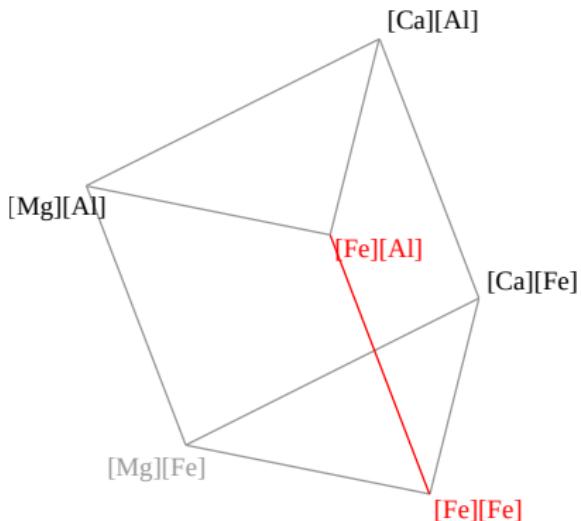
Finding R , μ and $Cov(\mu)$: R

- ▶ We seek a way to transform solid solutions so that they only include relevant endmembers.
- ▶ To the left, we see a prism that represents two-site $[Mg, Fe, Ca]_3[Al, Fe]_2Si_3O_{12}$ garnets.
- ▶ An independent endmember set (py-alm-gr-andr) is shown in **bold**.
- ▶ Dependent endmembers are defined by the properties and interactions between the independent endmembers.



Finding R , μ and $Cov(\mu)$: R

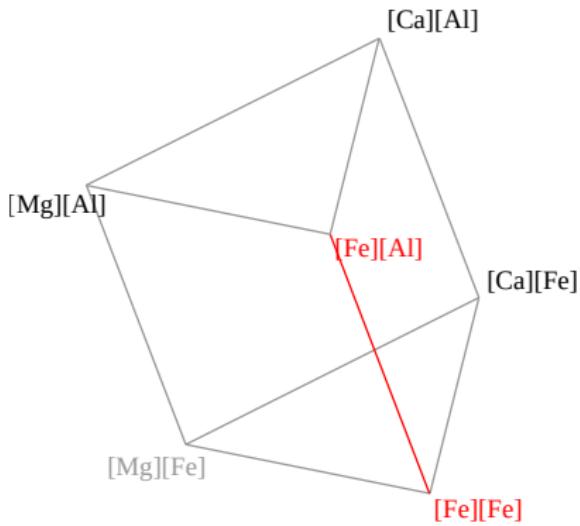
- ▶ Let's say we want to use experiments in the restricted system Fe-Al-Si-O.
- ▶ Pyrope, grossular and andradite are all unfeasible endmembers in this system, but the dependent endmember skiaigite ($[Fe]_3[Fe]_2Si_3O_{12}$) is feasible.
- ▶ Finding all the endmembers from an independent set is called vertex enumeration. The python module *cdd* (now in ENKI) does this.



Finding R , μ and $Cov(\mu)$: R

- ▶ After finding a feasible endmember basis for our system of interest, we can transform the original solution into that basis.
- ▶ For the
 $\text{py-alm-gr-andr} \longrightarrow \text{alm-sk}$
transformation, we type

```
FASO_garnet =  
solution_to_new_basis(  
    CFMASO_garnet,  
    [[0, 1, 0, 0], [0, 1, -1, 1]])
```
- ▶ After doing this for all solutions, we can find a set of well-constrained endmember reactions R ($\text{Ker}(S)$).



Finding R , μ and $Cov(\mu)$: $\mu(X, P, T)$

If endmember i is part of a solution, μ_i is dependent on the proportions p of endmembers in the solution. We can estimate p from analytical measurements using least squares, minimising the function:

$$\chi^2 = \left(M^T \cdot p - b \right) \cdot Cov(b)^{-1} \cdot \left(M^T \cdot p - b \right)^T$$

where M_{ij} is the molar amount of measured species j in each endmember i and b_j is the measured amount of species j . For low pressure CFMASO garnet, M looks like the following:

Endmember	Ca	FeT	Mg	Al	Si	Fe_B^{3+}
pyrope	0	0	3	2	3	0
almandine	0	3	0	2	3	0
grossular	3	0	0	2	3	0
andradite	3	2	0	0	3	2

Finding R , μ and $Cov(\mu)$: $Cov(\mu)$

Finally, we turn to defining the covariance matrix of μ . For any solution x , we propagate all uncertainties in composition in the standard way²:

$$Cov(\mu^x)_c = \left(\frac{\partial \mu_i^x}{\partial p_j^x} \right) \cdot Cov(p^x) \cdot \left(\frac{\partial \mu_i^x}{\partial p_j^x} \right)^T \quad (5)$$

We can combine these matrices into a block diagonal matrix for the whole assemblage:

$$Cov(\mu)_c = \begin{pmatrix} Cov(\mu^x)_c & 0 & 0 \\ 0 & Cov(\mu^y)_c & 0 \\ 0 & 0 & \ddots \end{pmatrix} \quad (6)$$

N.B. The partial derivative $\partial \mu_i^x / \partial p_j^x$ is also the compositional Hessian of the molar Gibbs free energy

Finding R , μ and $Cov(\mu)$: $Cov(\mu)$

Uncertainties in P and T ($z = [P, T]$) also contribute to $Cov(\mu)$:

$$Cov(\mu)_z = \left(\frac{\partial \mu}{\partial z} \right) \cdot Cov(z) \cdot \left(\frac{\partial \mu}{\partial z} \right)^T \quad (7)$$

The partial derivatives are the partial entropies and volumes of the endmembers (easily calculated using ENKI).

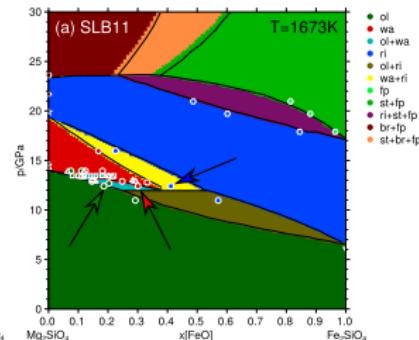
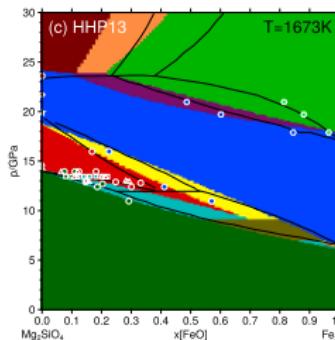
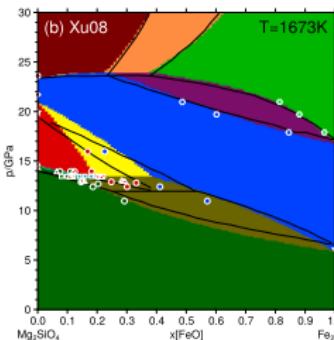
The total covariance is then taken to be equal to the sum of the components due to compositional and state uncertainties:

$$Cov(\mu) = Cov(\mu)_c + Cov(\mu)_z \quad (8)$$

Now we have defined R , μ and $Cov(\mu)$ for each assemblage, we can compute the likelihood function $p(D|\Theta)$.

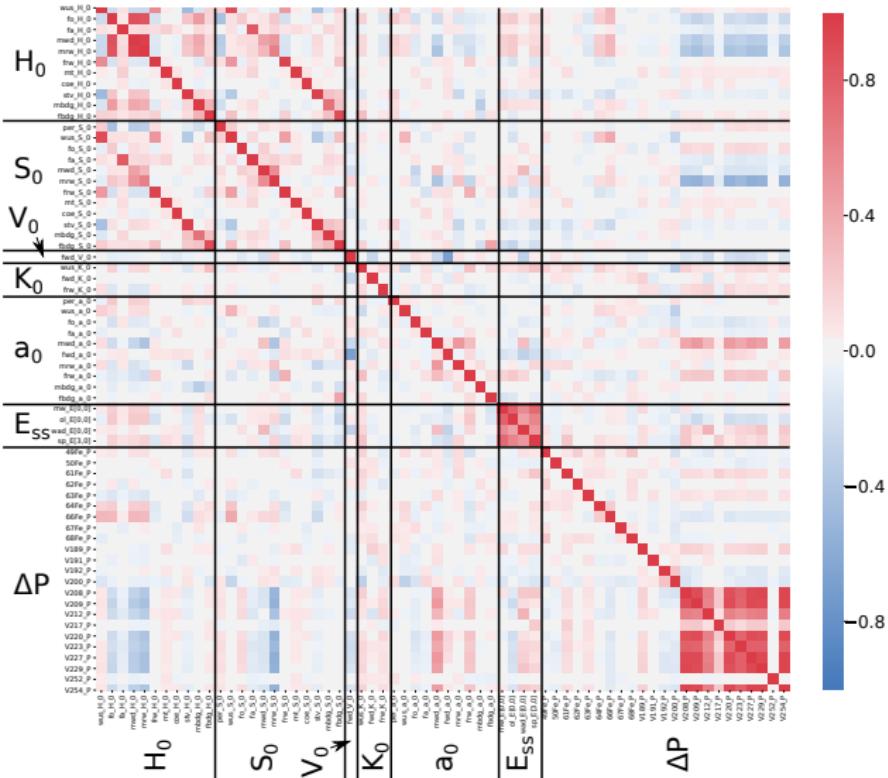
FMSO example

- ▶ We use the Fe-Mg-Si-O system as a test case.
- ▶ Our test suite includes 375 experimental assemblages
- ▶ We use the *emcee* python package, which implements an affine-invariant Markov-chain Monte Carlo. This is jargon, but the important points are that:
 - ▶ it's easy to use, and
 - ▶ works well when parameter values are poorly scaled and highly correlated



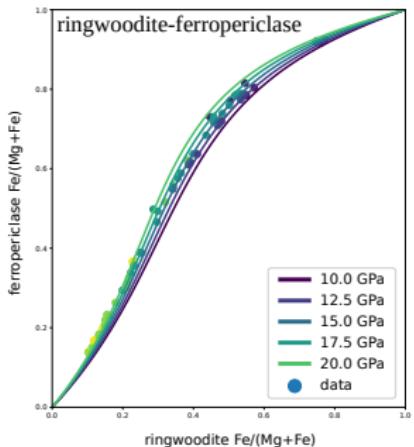
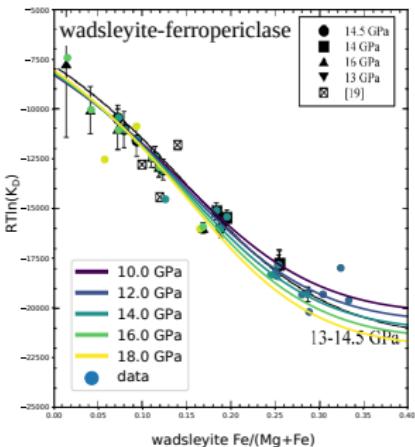
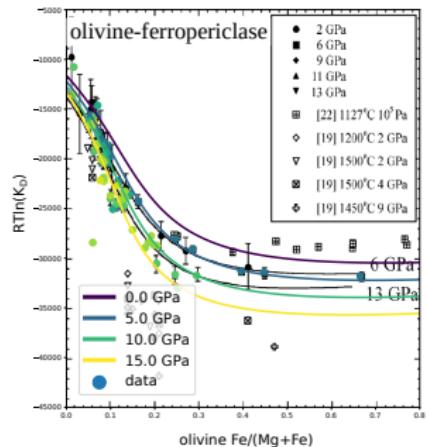
FMSO example: Parameter correlation matrix

- ▶ 62 parameters
- ▶ Some parameters highly correlated:
 - ▶ ol-wad-rw H_0
 - ▶ H_0 and S_0
 - ▶ Solution interactions
 - ▶ Pressure shifts!

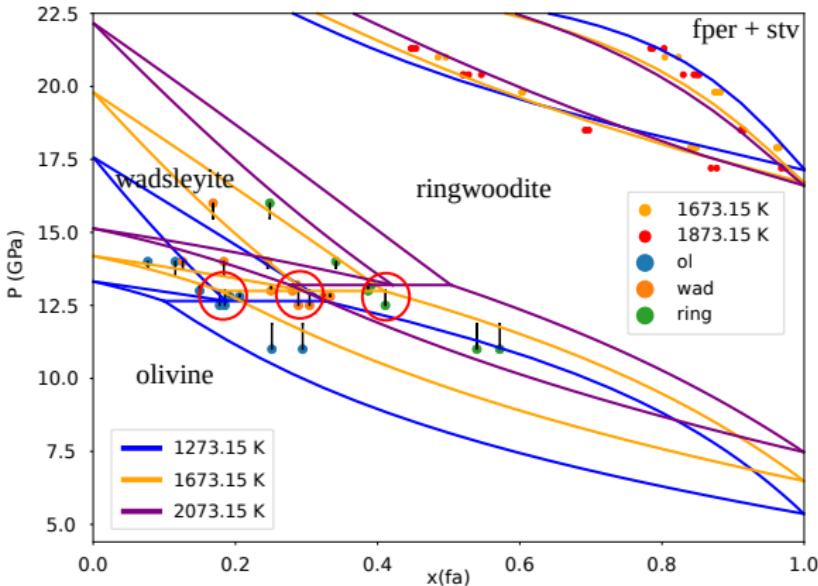


FMSO example: Mg-Fe partitioning

Partitioning experiments are accurately reproduced:



FMSO example: The fo-fa phase diagram



	fper	ol	wad	ring
Frost (2003)	11.1	2	7.5	4.16
This study	11.5 ± 0.1	2.3 ± 0.3	8.7 ± 0.5	4.2 ± 0.2
Jacobs et al. (2019)	13.4 ± 1.0	4.8 ± 0.5	4.43 ± 0.7	3.9 ± 0.6

Further work

- ▶ Extend to larger chemical systems
(especially ferric-iron-bearing systems)
- ▶ Incorporate kinetics/diffusion (a la Lucy, Aaron, Marc, Mark),
maintaining reaction-basis invariance (c.f. Bothe and Dreyer, 2014)
- ▶ Try creating a dataset using natural samples ☺
As long as there are > 2 reactions, even assemblages with unknown equilibration conditions provide useful thermodynamic information!

This slide left intentionally blank

The Jacobian of the equilibrium problem

The Jacobian matrix $\mathbf{J} = \partial \mathbf{F} / \partial \mathbf{x}$ is exceptionally useful both for solving the equilibrium problem, for creating starting guesses and for postprocessing (and probably other stuff too!).

$$\begin{aligned}\mathbf{F} &= [C_1 \quad C_2 \quad \mathcal{G}_r \quad \mathbf{B}]^T \\ \mathbf{x} &= [P \quad T \quad \mathbf{X}]\end{aligned}$$

Vectors \mathcal{G}_r and \mathbf{B} derive from the nullspace (multiplied by the endmember free energies) and row space of the transpose of the stoichiometric matrix. The vector \mathbf{X} contains all the phase composition information.

If we want to allow compositional variations, they can be added via modifications to the stoichiometric matrix. Essentially, an extra imaginary phase is added with a Gibbs free energy, entropy and volume of zero, and a formula corresponding to the compositional vector.

Bounded affine-invariant damped Newton root finding

Solver for the multivariate nonlinear system $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ with Jacobian $\mathbf{J}(\mathbf{x})$, using the damped affine invariant modification to Newton's method (Deuflhard, 1974;1975;2004). Here we follow the algorithm as described in Nowak and Weimann (1991): [Technical Report TR-91-10, Algorithm B], modified to accept linear inequality constraints.

Linear inequality constraints are also allowed. The constraints are satisfied if $\mathbf{A} \cdot \mathbf{x} + \mathbf{b} \leq \mathbf{0}$. If any constraints are not satisfied by the current damping value λ , λ is reduced to satisfy all the constraints.

Bounded damped affine-invariant Newton root finding

If a current iterate starting point (x_i) lies on one or more constraints and the Newton step violates one or more of those constraints, then the next step is calculated via the method of Lagrangian multipliers, minimizing the L2-norm of $F(x_i + \Delta)$ subject to the violated constraints.

Successful termination of the solver is based on three criteria:

- ▶ $\text{all}(\text{abs}(\text{dx (simplified newton step)}) < \text{tol}))$
- ▶ $\text{all}(\text{abs}(\text{dx (full Newton step)}) < \text{sqrt}(10*\text{tol})))$ [avoids pathological behaviour] and
- ▶ $\lambda = \lambda_{\max}(\text{dx}, x)$ ($\lambda = 1$ for a full Newton step).

The Clapeyron and Ehrenfest equations

The Clapeyron equation (for first order reactions)

$$\frac{\partial P}{\partial T} = -\frac{\Delta S}{\Delta V}$$

The first Ehrenfest equation (for second order reactions)

$$\frac{\partial P}{\partial T} = -\frac{\Delta \alpha}{\Delta \beta}$$